ELSEVIER

Signal Processing 42 (1995) 19–36

SIGNAL PROCESSING

A fast algorithm for $\mathcal{Q}\mathcal{R}^{-1}$ factorization of Toeplitz matrices

George-Othon Glentis

University of Twente, Department of Electrical Engineering, EL-BSC, Postbus 217, 7500 AE Enschede, The Netherlands

Received 18 February 1993; revised 23 December 1993, 14 April 1994 and 4 July 1994

Abstract

In this paper a new order recursive algorithm for the efficient $\mathcal{Q}\mathcal{R}^{-1}$ factorization of Toeplitz matrices is described. The proposed algorithm can be seen as a fast modified Gram–Schmidt method which recursively computes the orthonormal columns $\mathcal{Q}_i$, $i = 1, 2, \ldots, p$, of $\mathcal{Q}$, as well as the elements of $\mathcal{R}^{-1}$, of a Toeplitz matrix $\mathcal{X}$ with dimensions $L \times p$. The $\mathcal{Q}$ factor estimation requires $8Lp$ MADS (multiplications and divisions). Matrix $\mathcal{R}^{-1}$ is subsequently estimated using $3p^2$ MADS. A faster algorithm, based on a mixed $\mathcal{Q}$ and $\mathcal{R}^{-1}$ updating scheme, is also derived. It requires $7Lp + 3.5p^2$ MADS. The algorithm can be efficiently applied to batch least squares FIR filtering and system identification. When determination of the optimal filter is the desired task it can be utilized to compute the least squares filter in an order recursive way. The algorithm operates directly on the experimental data, overcoming the need for covariance estimates. An orthogonalized version of the proposed $\mathcal{Q}\mathcal{R}^{-1}$ algorithm is derived. Matlab code implementing the algorithm is also supplied.

Zusammenfassung

In dieser Arbeit wird ein neuer ordnungsrekursiver Algorithmus zur effizienten $\mathcal{Q}\mathcal{R}^{-1}$ Faktorisierung von Toeplitz-Matrizen beschrieben. Der vorgeschlagene Algorithmus kann als schnelle Modifizierte Gram–Schmidt-Methode betrachtet werden, bei der die orthonormalen Spalten $\mathcal{Q}_i$, $i = 1, 2, \ldots, p$, von $\mathcal{Q}$ sowie die Elemente von $\mathcal{R}^{-1}$ einer Toeplitz-Matrix $\mathcal{X}$ mit der Dimension $L \times p$ berechnet werden. Der $\mathcal{Q}$-Faktor wird mit einem Aufwand von $8Lp$ MADs (Multiplikationen und Divisionen) geschätzt. Die Matrix $\mathcal{R}^{-1}$ wird fortlaufend unter Anwendung von $3p^2$ MADs geschätzt. Weiterhin wird ein schnellerer Algorithmus abgeleitet, der auf einem gemischten $\mathcal{Q}$ und $\mathcal{R}^{-1}$ Aktualisierungsschema basiert. Er erfordert $7Lp + 3.5p^2$ MADs. Der Algorithmus kann effizient zur Echtzeit Least-Squares FIR-Filterung und Systemidentifkation benutzt werden. Wenn die Bestimmung des Optimalfilters verlangt ist, kann er zur Berechnung des Least-Squares-Filter in ordnungsrekursiver Form herangezogen werden. Der Algorithmus arbeitet direkt mit Hilfe der experimentellen Daten und vermeidet die Schätzung der Kovarianz. Es wird eine orthogonalisierte Version des vorgeschlagenen $\mathcal{Q}\mathcal{R}^{-1}$ Algorithmus abgeleitet. Eine Matlab Implementierung des Algorithmus wird angegeben.

Résumé

Dans cet article, un nouvel algorithme récursif d'ordre permettant une factorisation $\mathcal{Q}\mathcal{R}^{-1}$ efficace des matrices de Toeplitz est décrit. L'algorithme proposé peut être vu comme une méthode rapide de Gram–Schmidt Modifié laquelle calcule récursivement les colonnes orthonormales $\mathcal{Q}_i$, $i = 1, 2, \ldots, p$, de $\mathcal{Q}$ ainsi que les éléments de $\mathcal{R}^{-1}$ d'une matrice $\mathcal{X}$ de Toeplitz de dimension $L \times p$. Le facteur $\mathcal{Q}$ est estimé en terme de $8Lp$ MADS (Multiplications et Divisions). La matrice $\mathcal{R}^{-1}$ est estimée ultérieurement en utilisant le MADS $3p^2$. Un algorithme plus rapide basé sur un schéma de rafraichissement mixte de $\mathcal{Q}$ et de $\mathcal{R}^{-1}$ est également décrit. Il nécessite $7Lp + 3.5p^2$ MADS. L'algorithme peut être

SSDI 0165-1684(94)00113-8

efficacement appliqué pour le filtrage au moindre carrés FIR. Lorsque la détermination du filtre optimal est l'objectif désiré, il peut être utilisé pour calculer le filtre au moindre carré d'une manière récursive. L'algorithme utilise directement les données expérimentales, surmontant le besoin de l'estimation des covariances. Une version orthogonalisée de l'algorithme $\mathcal{LR}^{-1}$ est dérivée. L'implémentation Matlab du code est également donnée.

Keywords: Least squares estimation; QR factorization; Efficient algorithms

1. Introduction

In this paper a new order recursive algorithm for the efficient $\mathcal{LR}^{-1}$ factorization of a Toeplitz matrix and the determination of the least squares solution is developed. Both topics are of general importance in a wide range of applications. Typical examples include the design of optimal FIR filters, linear prediction of speech, time series analysis, orthogonal polynomials and Pade approximation [1, 14, 18, 21, 22, 24, 25]. $\mathcal{LR}$ algorithms have been successfully used in many signal processing applications for *adaptive* as well as for *batch* processing. Adaptive $\mathcal{LR}$ algorithms have been proposed for the efficient time update of the matrix factorization, taking into account the underlying Toeplitz property of the data matrix [7, 13, 17, 19, 20, 27, 33]. Efficient $\mathcal{LR}$ algorithms for batch (or block) processing have been derived in [4, 6, 8, 10–12, 15, 28, 29, 31], where in most cases, order updating schemes that take into account the Toeplitz structure are utilized.

The proposed $\mathcal{LR}^{-1}$ factorization algorithm is a batch processing technique. It applies to a block of data and operates in an order recursive way. From this perspective, it can be utilized for the batch least squares FIR filtering and system identification. It can be formulated in the following way. Given an input $x(n) \in \Re$ and a desired response $y(n) \in \Re$, over an observation interval $n \in [M, N]$, $M < N$. We seek to determine the optimal filter of *order* p, $\boldsymbol{c}_p = [c(i)]_{i=1,\ldots,p}$, that minimizes the error norm [1, 22, 24, 32]

$$\min_{\boldsymbol{c}_p} \| \Psi(M,N) + \mathcal{X}(M,N)\boldsymbol{c}_p \|_2, \tag{1}$$

where $\mathcal{X}(M,N)$ is the input data matrix of dimensions $(N-M-p+1)\times p$,

$$\mathcal{X}(M,N) = \begin{bmatrix} x(M+p-1) & x(M+p-2) & \cdots & x(M+1) & x(M) \\ x(M+p) & x(M+p-1) & \cdots & x(M+2) & x(M+1) \\ x(M+p+1) & x(M+p) & \cdots & x(M+3) & x(M+2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x(N-1) & x(N-2) & \cdots & x(N-p+1) & x(N-p) \\ x(N) & x(N-1) & \cdots & x(N-p+2) & x(N-p+1) \end{bmatrix}, \tag{2}$$

and $\Psi(M,N)$ is the desired response data vector of dimensions $(N-M-p+1)\times 1$,

$$\Psi(M,N) = [\, y(M+p-1)\ \ y(M+p)\ \ \ldots\ \ y(N-1)\ \ y(N)]^{\mathrm{T}}. \tag{3}$$

Matrix $\mathcal{X}(M,N)$ is Toeplitz. Minimization of (1) with respect to $\boldsymbol{c}_p$ leads to the linear system (4), the so-called normal equations

$$\mathcal{X}(M,N)^{\mathrm{T}}\mathcal{X}(M,N)\boldsymbol{c}_p(M,N) = -\mathcal{X}(M,N)\Psi(M,N). \tag{4}$$

The optimal filter of order p is then obtained as the solution of (4),

$$\boldsymbol{c}_p(M,N) = -(\mathcal{X}(M,N)^{\mathrm{T}}\mathcal{X}(M,N))^{-1}\mathcal{X}(M,N)\Psi(M,N), \tag{5}$$

and the residual error is given by the formula

$$E_p^c(M, N) = (\boldsymbol{I}_p - \mathscr{X}(M, N)(\mathscr{X}(M, N)^T \mathscr{X}(M, N))^{-1} \mathscr{X}(M, N))\, \Psi(M, N). \tag{6}$$

Eq. (5) provides an algorithm for computing the optimal parameter vector $\boldsymbol{c}_p$. The Toeplitz structure of $\mathscr{X}(M, N)$ gives rise to the use of efficient methods for the solution of the normal equations. Several fast, order recursive algorithms exist that solve (5) using $2Lp + O(p^2)$ MADS [5, 16, 23, 26]. Throughout this paper, abbreviation MADS means (M)ultiplications (A)nd (D)ivisions.

An alternative methodology for determining $\boldsymbol{c}_p$ is based on the $\mathscr{Q}\mathscr{R}$ factorization of the data matrix $\mathscr{X}(M, N)$. $\mathscr{Q}\mathscr{R}$ methods are discussed for their superior performance over the normal equations approach, especially in badly conditioned problems. Fast $\mathscr{Q}\mathscr{R}$ algorithms have been derived by taking into account the Toeplitz structure of $\mathscr{X}(M, N)$ [4, 6, 8, 10–12, 15, 28, 29, 31]. Algorithms [4, 6, 28, 31] are based on fast Cholesky factorization methods. In [10] an algorithm is derived, utilizing a Toeplitz embedding technique and the method described in [8]. It computes the $\mathscr{Q}$ factor and the inverse $\mathscr{R}^{-1}$ using $9Lp + 13.5p^2$ MADS with $L = N - M + p - 1$ being the leading dimension of $\mathscr{X}(M, N)$. Similarly, [12] uses a Toeplitz embedding and a generalized Levinson algorithm to derive a scheme for the factorization of the data matrix. It requires $7Lp + 7p^2$ MADS for the estimation of the $\mathscr{Q}$ factor. $\mathscr{R}^{-1}$ can subsequently be estimated using $3p^2$ MADS, thus resulting in an overall $7Lp + 10p^2$ MADS.

The proposed method operates directly on the data matrix. It performs a fast modified Gram–Schmidt algorithm on $\mathscr{X}(M, N)$. It recursively computes the orthonormal columns $\mathscr{Q}_i$, $i = 1, 2, \ldots, p$, of $\mathscr{Q}$ as well as the elements of $\mathscr{R}^{-1}$. When the determination of the optimal filter is the desired task it can be utilized to compute the least squares solution in an order recursive way. The computational complexity for the $\mathscr{Q}$ factor is $8Lp$ MADS. Subsequently, $\mathscr{R}^{-1}$ is estimated using $3p^2$ MADS. Alternatively, a mixed $\mathscr{Q}$ and $\mathscr{R}^{-1}$ updating scheme that requires $7Lp + 3.5p^2$ MADS is derived. Finally, when the estimation of the optimal filter $\boldsymbol{c}_p$ and the corresponding residual error is the desired task, $2Lp + 0.5p^2$ extra operations are required. Thus, the new algorithm is faster than existing ones. In the special case when $x(n) = 0$, $n < M + p - 1$ and $n > N - p$, the algorithm reduces to the classical lattice structure [8, 29]. An orthonormalized version of the method is also derived, which can be viewed as a generalization of the normalized lattice algorithm [29].

The main advantages of the proposed algorithm over those presented in [10, 12] are summarized as follows. First, orthogonalization is performed using the modified Gram–Schmidt method. Hence a nice numerical behavior is expected. Secondly, matrix embedding that is necessary in [10, 12] is bypassed, since operations are now performed directly on fixed size columns of data matrix $\mathscr{X}(M, N)$. Therefore, the computational complexity is minimized. Finally, the internal variables of the proposed orthonormalized version remain bounded, thus being suitable for real time implementation on fixed point processors or CORDIC architectures.

The paper is organized as follows. Definitions of the parameters used by the algorithm are given in Section 2. The order updates of the pertinent variables are derived in Section 3. The orthonormalized algorithm is presented in Section 4. The Matlab code implementing the orthogonalized algorithm is supplied in Appendix B.

2. $\mathscr{Q}\mathscr{R}$ factorization and the linear least squares problem

Let us define the vector $\boldsymbol{X}_m(M, N)$, of dimensions $(N - M - p + 1) \times 1$,

$$\boldsymbol{X}_m(M, N) = [x(M + p - m)\ \ x(M + p - 1 - m)\ \ \ldots\ \ x(N + 2 - m)\ \ x(N + 1 - m)]^T. \tag{7}$$

It represents the mth column of data matrix $\mathscr{X}(M, N)$. We then define the Toeplitz matrix $\mathscr{X}_m(M, N)$, consisting of the first m columns of $\mathscr{X}(M, N)$,

$$\mathscr{X}_m(M, N) = [\boldsymbol{X}_1(M, N)\ \boldsymbol{X}_2(M, N)\ \ \ldots\ \ \boldsymbol{X}_{m-1}(M, N)\ \boldsymbol{X}_m(M, N)]. \tag{8}$$

Clearly, $\mathscr{X}_p(M, N) = \mathscr{X}(M, N)$. The projection matrix $\mathscr{P}_{\mathscr{X}_m(M,N)}$ onto the subspace spanned by the columns of $\mathscr{X}_m(M, N)$ is then defined as [2, 3]

$$\mathscr{P}_{\mathscr{X}_m(M,N)} = \mathscr{X}_m(M, N)(\mathscr{X}_m^{\mathrm{T}}(M, N)\mathscr{X}_m(M, N))^{-1}\mathscr{X}_m^{\mathrm{T}}(M, N), \tag{9}$$

and the projection matrix onto the orthogonal complement of the column space of $\mathscr{X}_m(M, N)$ is expressed as

$$\mathscr{P}^{\perp}_{\mathscr{X}_m(M,N)} = \boldsymbol{I} - \mathscr{P}_{\mathscr{X}_m(M,N)}. \tag{10}$$

The Gram–Schmidt orthogonalization theory [2, 3] states that the orthogonal columns of the $\mathscr{Q}$ factor of matrix $\mathscr{X}(M, N)$ are obtained as

$$\begin{aligned}\mathscr{Q} &= [\mathscr{Q}_1, \mathscr{Q}_2, \ldots, \mathscr{Q}_p] \\ &= [\boldsymbol{X}_1(M, N), \mathscr{P}^{\perp}_{\mathscr{X}_1(M,N)}\boldsymbol{X}_2(M, N), \mathscr{P}^{\perp}_{\mathscr{X}_2(M,N)}\boldsymbol{X}_3(M, N), \ldots, \mathscr{P}^{\perp}_{\mathscr{X}_{p-1}(M,N)}\boldsymbol{X}_p(M, N)]\end{aligned} \tag{11}$$

or

$$\mathscr{Q}_1 = \boldsymbol{X}_1(M, N), \qquad \mathscr{Q}_{i+1} = \mathscr{P}^{\perp}_{\mathscr{X}_i(M,N)}\boldsymbol{X}_{i+1}(M, N), \quad i = 1, 2, \ldots, p-1.$$

The proposed algorithm provides an efficient way for the computation of orthogonal columns $\mathscr{Q}_i$, $i = 1, 2, \ldots, p-1$, using an order recursive scheme. A comparison of the computational flow graph between the fast $\mathscr{Q}\mathscr{R}^{-1}$ method and the modified Gram–Schmidt technique is depicted in Fig. 1.

The computation of the orthogonal columns $\mathscr{Q}_i$ is based on a set of residual error parameters that propagate through the algorithm. Reflection coefficients are then estimated as inner products of proper residual error parameters. They serve for the order recursive update of residual errors themselves, as well as for the recursive estimation of the least squares filters. Lower dimension filters and the corresponding residual errors, $\boldsymbol{c}_m, \boldsymbol{E}_m^c, \boldsymbol{b}_m, \boldsymbol{E}_m^b, \boldsymbol{a}_m, \boldsymbol{E}_m^f, \boldsymbol{w}_m, \boldsymbol{E}_m^w$, and $\boldsymbol{v}_m, \boldsymbol{E}_m^v$, $m = 1, 2, \ldots, p-1$, utilized by the algorithm are summarized in Table 1. They are all interpreted as least squares filters. Indeed, $\boldsymbol{c}_m(M, N)$, $m = 1, 2, \ldots, p$, are lower dimension least squares filters [32]. $\boldsymbol{a}_m(M, N)$ and $\boldsymbol{b}_m(M, N)$ are interpreted as optimal forward and backward predictors, respectively. The remaining two variables $\boldsymbol{w}_m(M, N)$ and $\boldsymbol{v}_m(M-1, N-1)$ may be

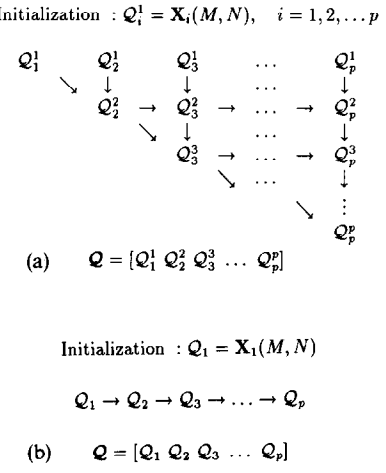

Fig. 1. (a) Computational flow graph for the Modified Gram–Schmidt algorithm [2]. Orthonormal columns of $\mathscr{Q}_{i+1}$ are successively estimated from the predecessors $\mathscr{Q}_i$ and a set of auxiliary column variables $\mathscr{Q}_i^j$, $1 \leqslant j \leqslant i$. (b) Computational flow graph for the proposed fast $\mathscr{Q}\mathscr{R}^{-1}$ algorithm. Orthonormal columns of $\mathscr{Q}$, $\mathscr{Q}_i = \mathscr{Q}_i^i$, $i = 1, 2, \ldots, p$, are recursively updated bypassing the need for auxiliary column variables $\mathscr{Q}_i^j$, $1 \leqslant j \leqslant i$, required by the Modified Gram–Schmidt algorithm.

Table 1
Least squares filters and residual error variables
(a)

Cost function	Filter
$\min_{c_m} \|\Psi(M,N) + \mathscr{X}_m(M,N)c_m\|$	$c_m(M,N) = -\mathscr{H}_{\mathscr{X}_m(M,N)}\Psi(M,N)$
$\min_{b_m} \|X_{m+1}(M,N) + \mathscr{X}_m(M,N)b_m\|$	$b_m(M,N) = -\mathscr{H}_{\mathscr{X}_m(M,N)}X_{m+1}(M,N)$
$\min_{a_m} \|X_1(M,N) + \mathscr{X}_m(M-1,N-1)a_m\|$	$a_m(M,N) = -\mathscr{H}_{\mathscr{X}_m(M-1,N-1)}X_1(M,N)$
$\min_{w_m} \|\delta^w + \mathscr{X}_m(M,N-1)w_m\|$	$w_m(M,N) = -\mathscr{H}_{\mathscr{X}_m(M,N-1)}\delta^w$
$\min_{v_m} \|\delta^v + \mathscr{X}_m(M-1,N-1)v_m\|$	$v_m(M-1,N-1) = -\mathscr{H}_{\mathscr{X}_m(M-1,N-1)}\delta^v$

(b)

Residual	Power
$E_m^c(M,N) = \mathscr{P}^{\perp}_{\mathscr{X}_m(M,N)}\Psi(M,N)$	$\alpha_m^c(M,N) = \langle E_m^c(M,N), E_m^c(M,N)\rangle$
$E_m^b(M,N) = \mathscr{P}^{\perp}_{\mathscr{X}_m(M,N)}X_{m+1}(M,N)$	$\alpha_m^b(M,N) = \langle E_m^b(M,N), E_m^b(M,N)\rangle$
$E_m^f(M,N) = \mathscr{P}^{\perp}_{\mathscr{X}_m(M-1,N-1)}X_1(M,N)$	$\alpha_m^f(M,N) = \langle E_m^f(M,N), E_m^f(M,N)\rangle$
$E_m^w(M,N) = \mathscr{P}^{\perp}_{\mathscr{X}_m(M,N-1)}\delta^w$	$\alpha_m^w(M,N) = \langle E_m^w(M,N), E_m^w(M,N)\rangle$
$E_m^v(M-1,N-1) = \mathscr{P}^{\perp}_{\mathscr{X}_m(M-1,N-1)}\delta^v$	$\alpha_m^v(M-1,N-1) = \langle E_m^q(M-1,N-1), E_m^v(M-1,N-1)\rangle$

Operator $\mathscr{H}_{\mathscr{X}_m(M,N)}$ is defined in terms of $\mathscr{X}_m(M,N)$ as $\mathscr{H}_{\mathscr{X}_m(M,N)} = \mathscr{X}_m(M,N)(\mathscr{X}_m^{\mathrm{T}}(M,N)\mathscr{X}_m(M,N))^{-1}$.

viewed as least squares filters taking as desired response the δ-signals $\delta^w = [0\ 0\ \dots\ 0\ 1]^{\mathrm{T}}$, and $\delta^v = [1\ 0\ \dots\ 0\ 0]^{\mathrm{T}}$, respectively, $\dim(\delta^w) = \dim(\delta^v) = (N-M-p+1)\times 1$.

A simple glance at Table 1 reveals that the orthogonal columns of $\mathscr{Q}$ and the backward prediction residuals $E_m^b(M,N)$ coincide. Thus, the $\mathscr{Q}$ factor of the Toeplitz matrix $\mathscr{X}(M,N)$ takes the form

$$\mathscr{Q}(M,N) = [X_1(M,N)\ E_1^b(M,N)\ \dots\ E_{p-1}^b(M,N)]. \tag{12}$$

Moreover,

$$\mathscr{X}(M,N)\mathscr{B}(M,N) = \mathscr{Q}(M,N), \tag{13}$$

where $\mathscr{B}(M,N)$ is an upper triangular matrix of dimensions $p\times p$, with diagonal elements having a value unity defined in terms of the elements of the backward predictors, $b_m(M,N)$,

$$\mathscr{B}(M,N) = \begin{bmatrix} 1 & b_1^1(M,N) & b_2^1(M,N) & \cdots & b_{p-1}^1(M,N) \\ 0 & 1 & b_2^2(M,N) & \cdots & b_{p-1}^2(M,N) \\ 0 & 0 & 1 & \cdots & b_{p-1}^3(M,N) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_{p-1}^{p-1}(M,N) \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{14}$$

Clearly, from Eqs. (13) and (14) we deduce that the $\mathscr{R}$ factor of $\mathscr{X}(M, N)$ is obtained as the inverse

$$\mathscr{R} = \mathscr{B}^{-1}(M, N) \tag{15}$$

or, equivalently, the inverse $\mathscr{R}^{-1}$ factor is recursively estimated as dictated by Eq. (14).

3. The algorithm

In this section we develop appropriate order recursions for the efficient update of the least squares filters and the pertinent residuals. In this way, the orthogonal columns of $\mathscr{Q}$ and the elements of $\mathscr{R}^{-1}$ are recursively computed. The derivation is based upon well known partitions of the increased order parameters that take into account the Toeplitz property of $\mathscr{X}_m(M, N)$.

Consider the increased order data matrix $\mathscr{X}_{m+1}(M, N)$, of dimensions $(N - M - p + 1) \times (m + 1)$. It is partitioned as

$$\mathscr{X}_{m+1}(M, N) = [\mathscr{X}_m(M, N)\ \boldsymbol{X}_{m+1}(M, N)] = [\boldsymbol{X}_1(M, N)\ \mathscr{X}_m(M - 1, N - 1)]. \tag{16}$$

The increased order sampled correlation matrix, of dimensions $(m + 1) \times (m + 1)$, is partitioned in the upper or in the lower form as

$$\begin{aligned}\boldsymbol{R}_{m+1}(M, N) &= \mathscr{X}_{m+1}^{\mathrm{T}}(M, N)\mathscr{X}_{m+1}(M, N) \\ &= \begin{bmatrix} \boldsymbol{R}_m(M, N) & \boldsymbol{r}_m^b(M, N) \\ \boldsymbol{r}_m^{b\mathrm{T}}(M, N) & r_m^{b0}(M, N) \end{bmatrix} = \begin{bmatrix} r_m^{f0}(M, N) & \boldsymbol{r}_m^f(M, N) \\ \boldsymbol{r}_m^{f\mathrm{T}}(M, N) & \boldsymbol{R}_m(M - 1, N - 1) \end{bmatrix},\end{aligned} \tag{17}$$

where

$$r_m^{b0}(M, N) = \langle \boldsymbol{X}_{m+1}(M, N), \boldsymbol{X}_{m+1}(M, N) \rangle, \qquad r_m^{f0}(M, N) = \langle \boldsymbol{X}_1(M, N), \boldsymbol{X}_1(M, N) \rangle. \tag{18}$$

We use notation $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$ to indicate the inner product of vectors $\boldsymbol{A}$ and $\boldsymbol{B}$, i.e. $\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \sum_{i=1}^{j} a(i) b(i)$, $j = \dim(\boldsymbol{A}) = \dim(\boldsymbol{B})$.

Similarly, the increased order sampled correlation vectors, $\boldsymbol{d}_{m+1}(M, N)$, $\boldsymbol{r}_{m+1}^b(M, N)$ and $\boldsymbol{r}_{m+1}^f(M, N)$, of dimensions $(m + 1) \times 1$, are partitioned as

$$\boldsymbol{d}_{m+1}(M, N) = \mathscr{X}_{m+1}^{\mathrm{T}}(M, N)\, \Psi(M, N) = \begin{bmatrix} \boldsymbol{d}_m(M, N) \\ d_{m+1}(M, N) \end{bmatrix}, \tag{19}$$

$$\boldsymbol{r}_{m+1}^b(M, N) = \mathscr{X}_{m+1}^{\mathrm{T}}(M, N)\boldsymbol{X}_{m+1}(M, N) = \begin{bmatrix} r_{m+1}^{fb}(M, N) \\ \boldsymbol{r}_{m+1}^b(M - 1, N - 1) \end{bmatrix}, \tag{20}$$

$$\boldsymbol{r}_{m+1}^f(M, N) = \mathscr{X}_{m+1}^{\mathrm{T}}(M - 1, N - 1)\boldsymbol{X}_1(M, N) = \begin{bmatrix} \boldsymbol{r}_m^{\mathrm{T}}(M, N) \\ r_{m+1}^{fb}(M, N) \end{bmatrix}, \tag{21}$$

where

$$d_{m+1}(M, N) = \langle \boldsymbol{X}_{m+1}(M, N), \Psi(M, N) \rangle, \qquad r_{m+1}^{fb}(M, N) = \langle \boldsymbol{X}_1(M, N), \boldsymbol{X}_{m+1}(M, N) \rangle. \tag{22}$$

Increased order parameters $\boldsymbol{c}_{m+1}$, $\boldsymbol{E}_{m+1}^c$, $\boldsymbol{b}_{m+1}$, $\boldsymbol{E}_{m+1}^b$, $\boldsymbol{a}_{m+1}$, $\boldsymbol{E}_{m+1}^f$, $\boldsymbol{w}_{m+1}$, $\boldsymbol{E}_{m+1}^w$ and $\boldsymbol{v}_{m+1}$, $\boldsymbol{E}_{m+1}^v$, are recursively computed on the basis of lower order counterparts. The proof is supplied in Appendix A, where order partitions of Eqs. (16)–(22) are coupled together with the matrix inversion lemma to produce an order updating scheme. The order recursive formulas for the estimation of the increased order parameters are summarized in Table 2. Notice that the error power variables α_m^b, α_m^f, α_m^w and α_m^v that are required by the algorithm can be efficiently computed in a recursive way [32], hereby bypassing the inner product operation.

Table 2
Order update recursions for the least squares filters and the corresponding residual errors

$$\boldsymbol{c}_{m+1}(M,N)=\begin{pmatrix}\boldsymbol{c}_m(M,N)\\0\end{pmatrix}+\begin{pmatrix}\boldsymbol{b}_m(M,N)\\1\end{pmatrix}k^c_{m+1}(M,N)$$
$$\boldsymbol{E}^c_{m+1}(M,N)=\boldsymbol{E}^c_m(M,N)+\boldsymbol{E}^b_m(M,N)\,k^c_{m+1}(M,N)$$
$$k^c_{m+1}(M,N)=-\beta^c_{m+1}(M,N)/\alpha^b_m(M,N)$$
$$\beta^c_{m+1}(M,N)=\langle\Psi(M,N),\boldsymbol{E}^b_m(M,N)\rangle$$

$$\boldsymbol{b}_{m+1}(M,N)=\begin{pmatrix}0\\\boldsymbol{b}_m(M-1,N-1)\end{pmatrix}+\begin{pmatrix}1\\\boldsymbol{a}_m(M,N)\end{pmatrix}k^b_{m+1}(M,N)$$
$$\boldsymbol{E}^b_{m+1}(M,N)=\boldsymbol{E}^b_m(M-1,N-1)+\boldsymbol{E}^f_m(M,N)\,k^b_{m+1}(M,N)$$
$$k^b_{m+1}(M,N)=-\beta^{fb}_{m+1}(M,N)/\alpha^f_m(M,N)$$
$$\beta^{fb}_{m+1}(M,N)=\langle\boldsymbol{X}_1(M,N),\boldsymbol{E}^b_m(M-1,N-1)\rangle$$
$$\alpha^f_{m+1}(M,N)=\alpha^f_m(M,N)+\beta^{fb}_{m+1}(M,N)k^f_{m+1}(M,N)$$

$$\boldsymbol{a}_{m+1}(M,N)=\begin{pmatrix}\boldsymbol{a}_m(M,N)\\0\end{pmatrix}+\begin{pmatrix}\boldsymbol{b}_m(M-1,N-1)\\1\end{pmatrix}k^f_{m+1}(M,N)$$
$$\boldsymbol{E}^f_{m+1}(M,N)=\boldsymbol{E}^f_m(M,N)+\boldsymbol{E}^b_m(M-1,N-1)\,k^f_{m+1}(M,N)$$
$$k^f_{m+1}(M,N)=-\beta^{fb}_{m+1}(M,N)/\alpha^b_m(M-1,N-1)$$
$$\alpha^b_{m+1}(M,N)=\alpha^b_m(M-1,N-1)+\beta^{fb}_{m+1}(M,N)k^b_{m+1}(M,N)$$

$$\boldsymbol{w}_{m+1}(M,N)=\begin{pmatrix}\boldsymbol{w}_m(M,N)\\0\end{pmatrix}+\begin{pmatrix}\boldsymbol{b}_m(M,N-1)\\1\end{pmatrix}k^w_{m+1}(M,N)$$
$$\boldsymbol{E}^w_{m+1}(M,N)=\boldsymbol{E}^w_m(M,N)+\boldsymbol{E}^b_m(M,N-1)\,k^w_{m+1}(M,N)$$
$$k^w_{m+1}(M,N)=\beta^w_{m+1}(M,N)/\alpha^b_m(M,N-1)$$
$$\beta^w_{m+1}(M,N)=\langle\boldsymbol{\delta}^w,\boldsymbol{E}^b_m(M,N-1)\rangle$$
$$\alpha^w_{m+1}(M,N)=\alpha^w_m(M,N)-\beta^w_{m+1}(M,N)k^w_{m+1}(M,N)$$

$$\boldsymbol{v}_{m+1}(M-1,N-1)=\begin{pmatrix}\boldsymbol{v}_m(M-1,N-1)\\0\end{pmatrix}+\begin{pmatrix}\boldsymbol{b}_m(M-1,N-1)\\1\end{pmatrix}k^v_{m+1}(M-1,N-1)$$
$$\boldsymbol{E}^v_{m+1}(M-1,N-1)=\boldsymbol{E}^v_m(M-1,N-1)+\boldsymbol{E}^b_m(M-1,N-1)\,k^v_{m+1}(M-1,N-1)$$
$$k^v_{m+1}(M-1,N-1)=-\beta^v_{m+1}(M-1,N-1)/\alpha^b_m(M-1,N-1)$$
$$\beta^v_{m+1}(M-1,N-1)=\langle\boldsymbol{\delta}^v,\boldsymbol{E}^b_m(M,N-1)\rangle$$
$$\alpha^v_{m+1}(M,N)=\alpha^v_m(M,N)+\beta^v_{m+1}(M,N)k^v_{m+1}(M,N)$$

The recursive scheme, developed so far for the computation of the $\mathscr{Q}$ and $\mathscr{R}^{-1}$ factors of the Toeplitz matrix $\mathscr{X}(M,N)$, as well as the computation of the least squares solution, requires the residual error vectors $\boldsymbol{E}^b_m(M,N-1)$ and $\boldsymbol{E}^b_m(M-1,N-1)$ as well as filters $\boldsymbol{b}_m(M,N-1)$ and $\boldsymbol{b}_m(M-1,N-1)$. The time shifts introduced can be accommodated using the matrix inversion lemma for modified matrices [16, 24]. Indeed, let us consider time shifted, sampled, autocorrelation matrices of order m,

$$\boldsymbol{R}_m(M,N)=\boldsymbol{R}_m(M,N-1)+\boldsymbol{x}_m(N)\boldsymbol{x}^{\mathrm{T}}_m(N), \tag{23}$$

$$\boldsymbol{R}_m(M-1,N-1)=\boldsymbol{R}_m(M,N-1)+\boldsymbol{x}_m(M+p-2)\boldsymbol{x}^{\mathrm{T}}_m(M+p-2), \tag{24}$$

where data vector $\boldsymbol{x}_m(n)$ is defined as $\boldsymbol{x}_m(n)=[x(n)\ x(n-1)\ \ldots\ x(n-m+1)]^{\mathrm{T}}$. The shifted, sampled, backward correlation vectors are accordingly modified as

$$\boldsymbol{r}^b_m(M,N)=\boldsymbol{r}^b_m(M,N-1)+\boldsymbol{x}_m(N)\,x(N-m), \tag{25}$$

$$\boldsymbol{r}^b_m(M-1,N-1)=\boldsymbol{r}^b_m(M,N-1)+\boldsymbol{x}_m(M+p-2)\,x(M+p-2-m). \tag{26}$$

Table 3
Time shifting updates for the backward predictor least squares filter and the corresponding residual error

$$\boldsymbol{b}_m(M, N-1) = \boldsymbol{b}_m(M, N) - \boldsymbol{w}_m(M, N)\varepsilon_m^b(N \mid M, N)$$

$$\boldsymbol{E}_m^b(M, N-1) = \begin{bmatrix} \mathscr{U}\boldsymbol{E}_m^b(M, N) - \mathscr{U}\boldsymbol{E}_m^w(M, N)\varepsilon_m^b(N \mid M, N) \\ \varepsilon_m^b(N \mid M, N-1) \end{bmatrix}$$

$$\varepsilon_m^b(N \mid M, N) = \langle \boldsymbol{\delta}^w, \boldsymbol{E}_m^b(M, N) \rangle,$$

$$\varepsilon_m^b(N \mid M, N-1) = e_m^b(N \mid M, N)\alpha_m^w(M, N)$$

$$\boldsymbol{b}_m(M-1, N-1) = \boldsymbol{b}_m(M, N-1) - \boldsymbol{v}_m(M-1, N-1)\varepsilon_m^b(M+p-2|M, N-1)$$

$$\boldsymbol{E}_m^b(M-1, N-1) = \begin{bmatrix} \varepsilon_m^b(M+p-2|M-1, N-1) \\ \mathscr{U}\boldsymbol{E}_m^b(M, N-1) - \mathscr{Z}\boldsymbol{E}_m^v(M-1, N-1)\varepsilon_m^b(M+p-2|M, N-1) \end{bmatrix}$$

$$\varepsilon_m^b(M+p-2|M-1, N-1) = \langle \boldsymbol{X}_{m+1}(M-1, N-1), \boldsymbol{E}_m^v(M-1, N-1) \rangle$$

$$\varepsilon_m^b(M+p-2|M, N-1) = e_m^b(M+p-2|M-1, N-1)/\alpha_m^v(M-1, N-1)$$

Repetitive application of the matrix inversion lemma for modified matrices leads to the time shifting recursions for the backward predictor and the corresponding residual error. They are tabulated in Table 3. Operators $\mathscr{U}$ and $\mathscr{Z}$ which appear in Table 3 are defined as

$$\mathscr{U} = [\boldsymbol{I}, \boldsymbol{0}], \qquad \mathscr{Z} = [\boldsymbol{0}, \boldsymbol{I}], \qquad \dim \boldsymbol{0} = (N - M - p + 1) \times 1.$$

Variables $\varepsilon_m^b(N \mid M, N)$ and $\varepsilon_m^b(N \mid M, N-1)$ appearing in Table 3 are interpreted as the backward a posteriori and a priori prediction errors at time instant N, based upon the backward predictor values $\boldsymbol{b}_m(M, N)$ and $\boldsymbol{b}_m(M, N-1)$, respectively. A posteriori and a priori backward prediction errors at time instant N are tied together as

$$\varepsilon_m^b(N \mid M, N-1) = \varepsilon_m^b(N \mid M, N)\alpha_m^w(M, N), \tag{27}$$

where $\alpha_m^w(M, N)$ is the error power of least squares filter $\boldsymbol{w}_m(M, N)$ defined in Table 1. Similarly, $\varepsilon_m^b(M+p-2 \mid M, N-1)$ and $\varepsilon_m^b(M+p-2 \mid M-1, N-1)$ can be viewed as a priori and a posteriori backward prediction errors, respectively. They are related by

$$\varepsilon_m^b(M+p-2 \mid M, N-1) = \varepsilon_m^b(M+p-2 \mid M-1, N-1)/\alpha_m^v(M-1, N-1), \tag{28}$$

where $\alpha_m^v(M-1, N-1)$ is the error power of least squares filter $\boldsymbol{v}_m(M-1, N-1)$ defined in Table 1. Moreover, it is easily seen that

$$\beta_{m+1}^w(M, N) = \varepsilon_m^b(N \mid M, N-1), \qquad \beta_{m+1}^v(M-1, N-1) = \varepsilon_m^b(M+p-2 \mid M-1, N-1).$$

The recursions developed so far for the order update of the least squares filters and the pertinent residuals are tied together to form a fast order recursive scheme for the efficient $\mathscr{Q}\mathscr{R}^{-1}$ decomposition of $\mathscr{X}(M, N)$ and the determination of the optimal solution. The reflection coefficients are first estimated from the lower order residual errors. They are subsequently utilized for the update of the residual errors themselves as well as for the update of the least squares filters.

The proposed algorithm for the efficient $\mathscr{Q}\mathscr{R}^{-1}$ factorization of Toeplitz matrices and the determination of the least squares solution, is tabulated in symbolic format in Table 4. To simplify notation we drop the order

Table 4
Fast $\mathscr{Q}\mathscr{R}^{-1}$ factorization algorithm with application to the estimation of the least squares FIR filter

INITIALIZATION

$\boldsymbol{E}^c = \Psi(M, N), \quad \boldsymbol{E}^b = \boldsymbol{E}^f = \boldsymbol{X}_1(M, N), \quad \boldsymbol{E}^w = \boldsymbol{\delta}^w, \quad \boldsymbol{E}^v = \boldsymbol{\delta}^v$
$\alpha^f = \alpha^b = \langle \boldsymbol{E}^f, \boldsymbol{E}^b \rangle$
$\alpha^w = \alpha^q = 1$

MAIN PART

FOR $m = 0$ **TO** $p-1$, **DO**		$\boldsymbol{E}^b - \boldsymbol{E}^b_{MN} + \boldsymbol{E}^f k^b$	L
$\varepsilon^{bw} = \langle \boldsymbol{\delta}^w, \boldsymbol{E}^b \rangle$	0	$\boldsymbol{E}^f = \boldsymbol{E}^f + \boldsymbol{E}^b_{MN} k^f$	L
$e^{bw} = \varepsilon^{bw} \alpha^w$	1	$\boldsymbol{E}^w = \boldsymbol{E}^w + \boldsymbol{E}^b_N k^w$	L
$\boldsymbol{E}^b_N = \begin{bmatrix} \mathscr{U}\boldsymbol{E}^b - \mathscr{U}\boldsymbol{E}^w \varepsilon^b \\ e^{bw} \end{bmatrix}$	$L-1$	$\boldsymbol{E}^v = \boldsymbol{E}^v + \boldsymbol{E}^b_{MN} k^v$	L
$\boldsymbol{b}_N = \boldsymbol{b} - \boldsymbol{w} e^{bw}$	m	$\boldsymbol{c} = \begin{pmatrix} \boldsymbol{c} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b} \\ 1 \end{pmatrix} k^c$	m
$\varepsilon^{bv} = \langle \boldsymbol{X}_{m+1}(M-1, N-1), \boldsymbol{E}^v \rangle$	L	$\boldsymbol{b} = \begin{pmatrix} 0 \\ \boldsymbol{b}_{MN} \end{pmatrix} + \begin{pmatrix} 1 \\ \boldsymbol{a} \end{pmatrix} k^b$	m
$e^{bv} = \varepsilon^{bv} / \alpha^v$	1	$\boldsymbol{a} = \begin{pmatrix} \boldsymbol{a} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}_{MN} \\ 1 \end{pmatrix} k^f$	m
$\boldsymbol{E}^b_{MN} = \begin{bmatrix} \varepsilon^{bv} \\ \mathscr{U}\boldsymbol{E}^b_N + \mathscr{L}\boldsymbol{E}^v e^{bv} \end{bmatrix}$	$L-1$	$\boldsymbol{w} = \begin{pmatrix} \boldsymbol{w} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}_N \\ 1 \end{pmatrix} k^w$	m
$\boldsymbol{b}_{MN} = \boldsymbol{b}_N + \boldsymbol{v} \varepsilon^{bv}$	m	$\boldsymbol{v} = \begin{pmatrix} \boldsymbol{v} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}_{MN} \\ 1 \end{pmatrix} k^c$	m
$\beta^c = \langle \boldsymbol{E}^b, \boldsymbol{E}^c \rangle$	L	$\alpha^b = \alpha^b_{MN} + \beta^{fb} k^b$	1
$\beta^{fb} = \langle \boldsymbol{E}^b_{MN}, \boldsymbol{E}^f \rangle$	L	$\alpha^f = \alpha^f + \beta^{fb} k^f$	1
$k^c = -\beta^c / \alpha^b$	1	$\alpha^w = \alpha^w - e^{bw} k^w$	1
$k^f = -\beta^c / \alpha^b_{MN}$	1	$\alpha^v = \alpha^v + \varepsilon^{bv} k^v$	1
$k^b = -\beta^{fb} / \alpha^f$	1	**END FOR** m	
$k^w = -e^{bw} / \alpha^b_N$	1	$\beta^c = \langle \boldsymbol{E}^b, \boldsymbol{E}^c \rangle$	L
$k^v = -\varepsilon^{bv} / \alpha^b_{MN}$	1	$k^c = -\beta^c / \alpha^b$	1
$\boldsymbol{E}^c = \boldsymbol{E}^c + \boldsymbol{E}^b k^c$	L	$\boldsymbol{c} = \begin{pmatrix} \boldsymbol{c} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b} \\ 1 \end{pmatrix} k^c$	$p-1$

END MAIN

and time indices. Boldface italic lower case letters $\boldsymbol{c}$, $\boldsymbol{a}$, $\boldsymbol{w}$, $\boldsymbol{v}$, $\boldsymbol{b}_N$, $\boldsymbol{b}_{MN}$, $\boldsymbol{r}$ and $\boldsymbol{d}$ are used to indicate vectors of varying dimensions, starting from 1, 2 and going up to $p-1$. Math italic characters are used to indicate scalar variables. For instance, vector $\boldsymbol{b}$ stands for the backward predictor $\boldsymbol{b}_m(M, N)$, while k^c represents $k^c_{m+1}(M, N)$. Boldface italic capital letters $\boldsymbol{E}^c, \boldsymbol{E}^b, \boldsymbol{E}^f, \boldsymbol{E}^w, \boldsymbol{E}^v, \boldsymbol{E}^b_N$ and $\boldsymbol{E}^b_{MN}$ represent the residual error vectors, of fixed dimensions $L \times 1$. $\boldsymbol{E}^c$, for example, stands for $\boldsymbol{E}^c_m(M, N)$. Subscript N denotes a time shift with respect to N, and subscript M denotes a time shift with respect to M, i.e. $\boldsymbol{b}_N$ denotes $\boldsymbol{b}_m(M, N-1)$ and $\boldsymbol{E}^b_{MN}$ stands for $\boldsymbol{E}^b_m(M-1, N-1)$.

The computational complexity of the fast $\mathcal{Q}\mathcal{R}^{-1}$ decomposition algorithm of Table 4 is $8Lp$ MADS for the $\mathcal{Q}$ factor and $3p^2$ for the $\mathcal{R}^{-1}$ part. A reduced complexity mixed updating scheme can also be derived, utilizing the identity

$$\begin{aligned}&\varepsilon_m^b(M+p-2\,|\,M-1,N-1)\\&\quad=\langle \boldsymbol{X}_{m+1}(M-1,N-1),\boldsymbol{E}_m^v(M-1,N-1)\rangle\\&\quad=(x(M+p-2-m)+\boldsymbol{x}_m^{\mathrm{T}}(M+p-2)\boldsymbol{v}_m(M-1,N-1))\alpha_m^v(M-1,N-1).\end{aligned}\tag{29}$$

The computation of the least squares filter of Eq. (1) and the corresponding residual error contribute to an additional amount of $2Lp+p^2$ MADS.

A Matlab function called `ls_fqr.m`, implementing the proposed algorithm for the efficient $\mathcal{Q}\mathcal{R}^{-1}$ factorization of Toeplitz matrices and the determination of the least squares solution, is supplied in Appendix B.

4. Orthonormalized $\mathcal{Q}\mathcal{R}^{-1}$ algorithm

The algorithms presented in the previous section for the efficient $\mathcal{Q}\mathcal{R}^{-1}$ factorization of the Toeplitz matrix $\mathcal{X}_p(M,N)$ can be viewed as orthogonalization methods. Orthogonal matrix $\mathcal{Q}$ may be directly normalized to produce the orthonormal matrix $\tilde{\mathcal{Q}}_p$ as

$$\tilde{\mathcal{Q}}_p=\mathcal{Q}_p(\mathcal{Q}_p^{\mathrm{T}}\mathcal{Q}_p)^{-1/2}\tag{30}$$

or

$$\tilde{\mathcal{Q}}_p=\mathcal{Q}_p\boldsymbol{D}^{-1/2},\tag{31}$$

where

$$\boldsymbol{D}^b=\mathrm{diag}\,[\alpha_0^b(M,N)\ \alpha_1^b(M,N)\ \ldots\ \alpha_{p-1}^b(M,N)].\tag{32}$$

Similarly,

$$\tilde{\mathcal{B}}_p=\mathcal{B}_p\boldsymbol{D}^{-1/2}.\tag{33}$$

An alternative method for determining the orthonormalized factorization of $\mathcal{X}_p(M,N)$ is presented in the sequel. A different internal parametrization is utilized and the computation of the orthonormalized columns of $\tilde{\mathcal{Q}}_p$ is carried out recursively. To introduce the method we define the so-called normalized filters and errors which are tabulated in Table 5(a).

Consider the order update recursion that computes the increased order backward filter, $\boldsymbol{b}_{m+1}(M,N)$ of Table 2. Using the normalized variables it is rewritten in the form

$$\begin{aligned}&\boldsymbol{B}_{m+1}(M,N)\alpha_{m+1}^{b/2}(M,N)\\&\quad=\begin{pmatrix}0\\ \boldsymbol{B}_m(M-1,N-1)\end{pmatrix}\alpha_m^{b/2}(M-1,N-1)+\begin{pmatrix}\boldsymbol{A}_m(M,N)\\ 0\end{pmatrix}\alpha_m^{f/2}(M,N)\,k_{m+1}^b(M,N)\end{aligned}\tag{34}$$

or

$$\boldsymbol{B}_{m+1}(M,N)=P_{m+1}^{fb}(M,N)\left(\begin{pmatrix}0\\ \boldsymbol{B}_m(M-1,N-1)\end{pmatrix}+\begin{pmatrix}\boldsymbol{A}_m(M,N)\\ 0\end{pmatrix}\tilde{k}_{m+1}^{fb}(M,N)\right),\tag{35}$$

Table 5
Normalized least squares filters and residual errors, normalized reflection coefficients and the corresponding scaling factors
(a)

Nomalized filters	Normalized residuals
$\boldsymbol{C}_m(M,N)=\begin{pmatrix}1\\ \boldsymbol{c}_m(M,N)\end{pmatrix}\alpha_m^{-c/2}(M,N)$	$\mathscr{E}_m^c(M,N)=\boldsymbol{E}_m^c(M,N)\alpha_m^{-c/2}(M,N)$
$\boldsymbol{B}_m(M,N)=\begin{pmatrix}\boldsymbol{b}_m(M,N)\\ 1\end{pmatrix}\alpha_m^{-b/2}(M,N)$	$\mathscr{E}_m^b(M,N)=\boldsymbol{E}_m^b(M,N)\alpha_m^{-b/2}(M,N)$
$\boldsymbol{A}_m(M,N)=\begin{pmatrix}1\\ \boldsymbol{a}_m(M,N)\end{pmatrix}\alpha_m^{-f/2}(M,N)$	$\mathscr{E}_m^f(M,N)=\boldsymbol{E}_m^f(M,N)\alpha_m^{-f/2}(M,N)$
$\boldsymbol{W}_m(M,N)=\begin{pmatrix}1\\ \boldsymbol{w}_m(M,N)\end{pmatrix}\alpha_m^{-w/2}(M,N)$	$\mathscr{E}_m^w(M,N)=\boldsymbol{E}_m^w(M,N)\alpha_m^{-w/2}(M,N)$
$\boldsymbol{V}_m(M-1,N-1)=\begin{pmatrix}1\\ \boldsymbol{v}_m(M-1,N-1)\end{pmatrix}\alpha_m^{-v/2}(M-1,N-1)$	$\mathscr{E}_m^v(M-1,N-1)=\boldsymbol{E}_m^v(M-1,N-1)\alpha_m^{-v/2}(M-1,N-1)$

(b)

Reflection coefficients	Scaling factors
$\tilde{k}_{m+1}^c(M,N)=\langle\mathscr{E}_m^b(M,N),\mathscr{E}_m^c(M,N)\rangle$	$P_{m+1}^c(M,N)=(1-\tilde{k}_{m+1}^{c2}(M,N))^{-1/2}$
$\tilde{k}_{m+1}^{fb}(M,N)=\langle\mathscr{E}_m^b(M-1,N-1),\mathscr{E}_m^f(M,N)\rangle$	$P_{m+1}^{fb}(M,N)=(1-\tilde{k}_{m+1}^{fb2}(M,N))^{-1/2}$
$\tilde{k}_{m+1}^w(M,N)=\langle\mathscr{E}_m^b(M,N-1),\mathscr{E}_m^w(M,N)\rangle$	$P_{m+1}^w(M,N)=(1+\tilde{k}_{m+1}^{w2}(M,N))^{-1/2}$
$\tilde{k}_{m+1}^v(M-1,N-1)=\langle\mathscr{E}_m^b(M-1,N-1),\mathscr{E}_m^v(M-1,N-1)\rangle$	$P_{m+1}^v(M-1,N-1)=(1-\tilde{k}_{m+1}^{v2}(M-1,N-1))^{-1/2}$

where

$$P_{m+1}^{fb}(M,N)=\alpha_{m+1}^{-b/2}(M,N)\alpha_m^{b/2}(M-1,N-1) \tag{36}$$

and

$$\tilde{k}_{m+1}^{fb}(M,N)=\alpha_m^{f/2}(M,N)k_{m+1}^b(M,N)\alpha_m^{-b/2}(M-1,N-1). \tag{37}$$

The normalized backward error variables are updated in a similar way as

$$\mathscr{E}_{m+1}^b(M,N)=P_{m+1}^{fb}(M,N)(\mathscr{E}_m^b(M-1,N-1)+\mathscr{E}_m^f(M,N)\tilde{k}_{m+1}^{fb}(M,N)). \tag{38}$$

Recalling that $\boldsymbol{E}_m^f(M,N)$ is orthogonal to $\boldsymbol{E}_{m+1}^b(M,N)$, and thus $\langle\mathscr{E}_{m+1}^b(M,N),\mathscr{E}_m^f(M,N)\rangle=0$, we get

$$\tilde{k}_{m+1}^{fb}(M,N)=-\langle\mathscr{E}_m^b(M-1,N-1),\mathscr{E}_m^f(M,N)\rangle. \tag{39}$$

Eq. (37) yields

$$\tilde{k}_{m+1}^{fb}(M,N)=-\alpha_{m+1}^{-f/2}(M,N)\beta_{m+1}^{fb}(M,N)\alpha_m^{-b/2}(M-1,N-1) \tag{40}$$

Table 6
Fast orthonormalized $\mathscr{Q}\mathscr{R}^{-1}$ factorization algorithm with application to the estimation of the least squares FIR filter

INITIALIZATION

$\alpha^c = \langle \Psi(M,N), \Psi(M,N)\rangle^{-1/2}$	L
$\mathscr{E}^c = \Psi(M,N)\alpha^c$	L
$\alpha^b = \alpha^f = \langle \boldsymbol{X}_1(M,N), \boldsymbol{X}_1(M,N)\rangle^{-1/2}$	L
$\mathscr{E}^b = \mathscr{E}^f = \boldsymbol{X}_1(M,N)\alpha^b$	L
$\mathscr{E}^w = \delta^w, \quad \mathscr{E}^v = \delta^v, \quad \alpha^w = \alpha^v = 1$	

MAIN PART

FOR $m = 0$ **TO** $p-1$, **DO**		$P^w = (1 - \tilde{k}^{w2})^{-1/2}$	1
$\tilde{\varepsilon}^{bw} = \langle \delta^w, \mathscr{E}^b\rangle$	0	$P^v = (1 - \tilde{k}^{v2})^{-1/2}$	1
$\tilde{e}^{bw} = \tilde{\varepsilon}^{bw}\alpha^w$	1	$\mathscr{E}^c = P^c(\mathscr{E}^c + \mathscr{E}^b\tilde{k}^c)$	$2L$
$P^{bw} = (1 - \tilde{e}^{bw2})^{-1/2}$	1	$\mathscr{E}^b = P^{fb}(\mathscr{E}^b_{MN} + \mathscr{E}^f\tilde{k}^{fb})$	$2L$
$\mathscr{E}^b_N = P^{bw}\begin{bmatrix}\mathscr{U}\mathscr{E}^b - \mathscr{U}\mathscr{E}^w\tilde{\varepsilon}^b \\ \tilde{e}^{b2}\end{bmatrix}$	$2L-1$	$\mathscr{E}^f = P^{fb}(\mathscr{E}^f + \mathscr{E}^b_{MN}\tilde{k}^{fb})$	$2L$
$\boldsymbol{B}_N = P^{bw}(\boldsymbol{B} - \boldsymbol{W}e^{bw})$	$2m$	$\mathscr{E}^w = P^w(\mathscr{E}^w + \mathscr{E}^b_N\tilde{k}^w)$	$2L$
$\tilde{\varepsilon}^{bv} = \langle \boldsymbol{X}_{m+1}(M,N), \mathscr{E}^v\rangle$	L	$\mathscr{E}^v = P^v(\mathscr{E}^v + \mathscr{E}^b_{MN}\tilde{k}^v)$	$2L$
$\tilde{e}^{bv} = \tilde{\varepsilon}^{bv}/\alpha^b_N$	1	$\boldsymbol{C} = P^c\left(\begin{pmatrix}\boldsymbol{C}\\0\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}\\1\end{pmatrix}\tilde{k}^c\right)$	$2m$
$P^{bv} = (1 - \tilde{\varepsilon}^{bv2})^{-1/2}$	1	$\boldsymbol{B} = P^{fb}\left(\begin{pmatrix}0\\\boldsymbol{B}_{MN}\end{pmatrix} + \begin{pmatrix}1\\\boldsymbol{A}\end{pmatrix}\tilde{k}^{fb}\right)$	$2m$
$\mathscr{E}^b_{MN} = P^{bv}\begin{bmatrix}\tilde{\varepsilon}^{bv2} \\ \mathscr{U}\mathscr{E}^b_N + \mathscr{Z}\mathscr{E}^v\tilde{e}^{bv}\end{bmatrix}$	$2L-1$	$\boldsymbol{A} = P^{fb}\left(\begin{pmatrix}\boldsymbol{A}\\0\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}_{MN}\\1\end{pmatrix}\tilde{k}^{fb}\right)$	$2m$
$\boldsymbol{B}_{MN} = P^{bv}(\boldsymbol{B}_N + \boldsymbol{V}\tilde{\varepsilon}^{bv})$	$2m$	$\boldsymbol{W} = P^w\left(\begin{pmatrix}\boldsymbol{W}\\0\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}_N\\\boldsymbol{I}\end{pmatrix}\tilde{k}^w\right)$	$2m$
$\tilde{k}^c = -\langle \mathscr{E}^b, \mathscr{E}^c\rangle$	L	$\boldsymbol{V} = P^v\left(\begin{pmatrix}\boldsymbol{V}\\0\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}_{MN}\\\boldsymbol{I}\end{pmatrix}\tilde{k}^v\right)$	$2m$
$\tilde{k}^{fb} = -\langle \mathscr{E}^b_{MN}, \mathscr{E}^f\rangle$	L	**END FOR** m	
$P^c = (1 - \tilde{k}^{c2})^{-1/2}$	1	$\tilde{k}^c = \langle \mathscr{E}^b, \mathscr{E}^c\rangle$	L
$P^{fb} = (1 - \tilde{k}^{fb2})^{-1/2}$	1	$\boldsymbol{C} = P^c\left(\begin{pmatrix}\boldsymbol{C}\\0\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}\\\boldsymbol{I}\end{pmatrix}\tilde{k}^c\right)$	$2p-1$

END MAIN

or

$$(\tilde{k}^{fb}_{m+1}(M,N))^2 = k^f_{m+1}(M,N)\,k^b_{m+1}(M,N). \tag{41}$$

Thus,

$$P^{fb}_{m+1}(M,N) = (1 - (\tilde{k}^{fb}_{m+1}(M,N))^2)^{-1/2}. \tag{42}$$

A similar argument leads to recursive equations for the remaining normalized filters and residuals. The formulas for the corresponding reflection coefficients and the scaling factors are tabulated in Table 5(b).

The resulting orthonormalized algorithm is presented in symbolic format in Table 6. It can be viewed as the generalization of the normalized lattice algorithm presented and analyzed [9, 30].

5. Conclusions

A fast algorithm for the $\mathscr{Q}\mathscr{R}^{-1}$ decomposition of a Toeplitz matrix $\mathscr{X}(M, N)$ has been presented. The computational complexity of the proposed method is $8Lp + 3p^2$ MADS, where L is the leading dimension of $\mathscr{X}(M, N)$. A reduced complexity mixed scheme that requires $7Lp + 3.5p^2$ MADS has also been derived. The least squares filter and the corresponding residual error can be computed using $2Lp + 0.5p^2$ additional operations. The algorithm can be efficiently applied to batch least squares FIR filtering and system identification. The orthonormalized version proposed uses approximately twice as much MADS and $O(p)$ square root operations.

Acknowledgements

The author would like to thank the reviewers for their constructive suggestions.

Appendix A. Proof of main recursions

A comprehensive proof of the basic formula utilized by the algorithm is supplied. Two basic recursions are analyzed, a two term Levinson type formula that updates filter vectors of increasing dimensions starting from 1, 2 and going up to $p - 1$, and a two term lattice type formula that computes residual error vectors of fixed dimensions $L \times 1$ (see Table 2).

We will concentrate on the order recursions for the least squares filter and the corresponding residual error. The remaining recursions summarized in Table 2 are proved in a similar way. The increased order least squares filter $\boldsymbol{c}_{m+1}(M, N)$ of dimensions $(m + 1) \times 1$ is obtained as the solution of the increased order linear system

$$\boldsymbol{R}_{m+1}(M, N)\boldsymbol{c}_{m+1}(M, N) = -\boldsymbol{d}_{m+1}(M, N), \tag{A.1}$$

or equivalently by applying the increased order operator $\mathscr{K}_{\mathscr{X}_{m+1}(M,N)}$ onto the data vector $\Psi(M, N)$ (see Table 1),

$$\boldsymbol{c}_{m+1}(M, N) = -\mathscr{K}^{\mathrm{T}}_{\mathscr{X}_{m+1}(M,N)}\Psi(M, N), \tag{A.2}$$

where

$$\mathscr{K}_{\mathscr{X}_{m+1}(M,N)} = \mathscr{X}_{m+1}(M, N)\boldsymbol{R}^{-1}_{m+1}(M, N). \tag{A.3}$$

Taking into account partition (16), the matrix inversion lemma for partitioned matrices and definitions of Table 1, we obtain

$$\mathscr{K}_{\mathscr{X}_{m+1}(M,N)} = [\mathscr{X}_m(M,N)\ \boldsymbol{X}_{m+1}(M,N)] \begin{bmatrix} \boldsymbol{R}_m^{-1}(M,N) + \dfrac{\boldsymbol{b}_m(M,N)\boldsymbol{b}_m^{\mathrm{T}}(M,N)}{\alpha_m^b(M,N)} & \dfrac{\boldsymbol{b}_m(M,N)}{\alpha_m^b(M,N)} \\ \dfrac{\boldsymbol{b}_m^{\mathrm{T}}(M,N)}{\alpha_m^b(M,N)} & \dfrac{1}{\alpha_m^b(M,N)} \end{bmatrix}$$

or

$$\mathscr{K}_{\mathscr{X}_{m+1}(M,N)} = \begin{bmatrix} \mathscr{K}_{\mathscr{X}_m(M,N)} + \dfrac{\boldsymbol{E}_m^b(M,N)}{\alpha_m^b(M,N)}\boldsymbol{b}_m^{\mathrm{T}}(M,N) & \dfrac{\boldsymbol{E}_m^b(M,N)}{\alpha_m^b(M,N)} \end{bmatrix}. \tag{A.4}$$

Application of (A.4) to (A.2) results in

$$\boldsymbol{c}_{m+1}(M,N) = \begin{pmatrix} \boldsymbol{c}_m(M,N) \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}_m(M,N) \\ 1 \end{pmatrix} k_{m+1}^c(M,N), \tag{A.5}$$

where

$$k_{m+1}^c(M,N) = -\frac{\beta_{m+1}^c(M,N)}{\alpha_m^b(M,N)}$$

and

$$\beta_{m+1}^c(M,N) = \langle \Psi(M,N), \boldsymbol{E}_m^b(M,N)\rangle = \langle \boldsymbol{E}_M^c(M,N), \boldsymbol{E}_m^b(M,N)\rangle.$$

The increased order lattice type recursion for the filter residual is obtained as

$$\boldsymbol{E}_{m+1}^c(M,N) = \mathscr{P}^{\perp}_{\mathscr{X}_{m+1}(M,N)}\Psi(M,N). \tag{A.6}$$

Operator $\mathscr{P}^{\perp}_{\mathscr{X}_{m+1}(M,N)}$ given by (10) can be written in terms of $\mathscr{K}_{\mathscr{X}_{m+1}(M,N)}$ as

$$\mathscr{P}^{\perp}_{\mathscr{X}_{m+1}(M,N)} = \boldsymbol{I} - \mathscr{K}_{\mathscr{X}_{m+1}(M,N)}\mathscr{X}_{m+1}^{\mathrm{T}}(M,N). \tag{A.7}$$

Applying partition (16) and (A.4) we obtain

$$\mathscr{P}^{\perp}_{\mathscr{X}_{m+1}(M,N)} = \boldsymbol{I} - \begin{bmatrix} \mathscr{K}_{\mathscr{X}_{m+1}(M,N)} + \dfrac{\boldsymbol{E}_m^b(M,N)}{\alpha_m^b(M,N)}\boldsymbol{b}_m^{\mathrm{T}}(M,N) & \dfrac{\boldsymbol{E}_m^b(M,N)}{\alpha_m^b(M,N)} \end{bmatrix} \begin{bmatrix} \mathscr{X}_m^{\mathrm{T}}(M,N) \\ \boldsymbol{X}_{m+1}^{\mathrm{T}}(M,N) \end{bmatrix} \tag{A.8}$$

or

$$\mathscr{P}^{\perp}_{\mathscr{X}_{m+1}(M,N)} = \mathscr{P}^{\perp}_{\mathscr{X}_m(M,N)} - \frac{\boldsymbol{E}_m^b(M,N)\boldsymbol{E}_m^{b\mathrm{T}}(M,N)}{\alpha_m^b(M,N)}. \tag{A.9}$$

Finally, Eqs. (A.6) and (A.9) result in

$$\boldsymbol{E}_{m+1}^c(M,N) = \boldsymbol{E}_m^c(M,N) + \boldsymbol{E}_m^b(M,N)k_{m+1}^c(M,N).$$

Appendix B. Matlab function ls_fqr.m

```
function [Q,B,c,Ec,ac,Aux_LS,Aux_E,Aux_pow]=ls_fqr(x_col,x_row,z)
%
%    FAST ALGORITHM FOR QR FACTORIZATION OF TOEPLITZ MATRICES
%
%         by    G.O. GLENTIS,    December 1993
%
%    Function Name: ls_fqr (Least Squares Fast QR)
%
%    Least squares solution on :  toeplitz(x_col,x_row)*c=-z
%    by performing a fast QR^{-1} factorization.
% ------------------------------------------------------- DIMENSIONS --
%    Inputs (3)  x_col : first column of Toeplitz matrix   (N-M-p+1) x 1
%                x_row : first row of Toeplitz matrix             p x 1
%                z     : desired response vector           (N-M-p+1) x 1
%
%   Outputs (12) Q     : orthogonal matrix factor          (N-M-p+1) x p
%                B     : inverse R factor                         p x p
%                c     : least squares solution                   p x 1
%                Ec    : least squares residual            (N-M-p+1) x 1
%     Aux_LS=[a b w v] : Auxiliary parameters (LS forward       (p-1) x 4
%              and backward predictors and pinning filters)
%  Aux_E=[Ef Eb Ew Ev] : Auxiliary parameters residuals     (N-M-p+1) x 4
% Aux_pow=[af ab aw av]: Auxiliary parameters powers                1 x 4
%
%
N=max(size(x_col)); p=max(size(x_row));
aw=1.; av=1.; ab=x_col'*x_col; af=ab; ac=z'*z;
Ef=x_col; Eb=x_col; Ew=[zeros(N-1,1); 1]; Ev=[1; zeros(N-1,1)];Ec=z;
Q=x_col; B=[1;zeros(p-1,1)];
c=[]; b=[]; a=[]; w=[]; v=[];
%
%   Main part
%
for i=1:p-1
%
% Time shifting of the backward residual errors
%
 ebn=Eb(N);
```

```
 Ebn(1:N-1,1)=Eb(1:N-1)-Ew(1:N-1)*ebn;
 Ebn(N)=ebn*aw;
 x_aux=[x_row(i+1:-1:2);x_col(1:N-i)];
 Ebmn(1)=x_aux'*Ev;
 ebm=Ebmn(1)/av;
 Ebmn(2:N,1)=Ebn(1:N-1)+Ev(2:N)*ebm;
 abn=ab-ebn*Ebn(N);
 abm=abn+ebm*Ebmn(1);
%
% Estimation of the reflection coefficients
%
 bf=Ef'*Ebmn;
 bc=Ec'*Eb;
 kc=-bc/ab;
 kb=-bf/af;
 kf=-bf/abm;
 kw=-Ebn(N)/abn;
 kv=-Ebmn(1)/abm;
%
% Order updates of the residual errors
%
 Ec=Ec+Eb*kc;
 Eb=Ebmn+Ef*kb;
 Ef=Ef+Ebmn*kf;
 Ew=Ew+Ebn*kw;
 Ev=Ev+Ebmn*kv;
%
% Order updates of the error powers
%
 ab=abm+bf*kb;
 af=af+bf*kf;
 aw=aw-Ebn(N)*kw;
 av=av+Ebmn(1)*kv;
 ac=ac+bc*kc;
%
% Time shifting of the backward predictor
%
 bn=b-w*ebn;
 bmn=bn+v*ebm;
%
```

```
% Order updates of the least squares filters
%
 c=[c; 0]+[b; 1]*kc;
 b=[0; bmn]+[1; a]*kb;
 a=[a; 0]+[bmn; 1]*kf;
 w=[w; 0]+[bn; 1]*kw;
 v=[v; 0]+[bmn; 1]*kv;
%
% Construction of the Q factor and the R inverse
%
 Q=[Q Eb];
 B=[ B [ b; 1; zeros(p-i-1,1)]];
end
%
% Last update for filter estimation
%
 bc=Ec'*Eb;
 kc=-bc/ab;
 c=[c; 0]+[b; 1]*kc;
 ac=ac+bc*kc;
%
% Prepare output
%
 Aux_LS=[a b w v];
 Aux_E=[Ef Eb Ew Ev];
 Aux_pow=[af ab aw av];
%
%   ------------------   END OF  ls_fqr.m   --------------------
%
```

References

[1] S.T. Alexander, *Adaptive Signal Processing, Theory and Applications*, Springer, New York, 1986.

[2] A. Bjorck, "Solving linear least squares problems by Gram–Schmidt orthogonalization", *BIT*, Vol. 7, 1967. pp. 1–21.

[3] A. Bjorck, Algorithms for linear least squares problems, NATO ASI in Computer Algorithms for Solving Linear Algebraic Equations, Series F, Vol. 77, 1990.

[4] A.W. Bojanczyk, R.P. Brent and F.R. de Hoog, "QR factorization of Toeplitz matrices", *Numer. Math.*, Vol. 49, 1986, pp. 81–94.

[5] G. Carayannis, D. Manolakis and N. Kalouptsidis, "A unified view of parametric processing algorithms for prewindowed signals", *Signal Processing*, Vol. 10, No. 4, June 1986, pp. 335–368.

[6] J. Chun, T. Kailath and H. Lev-Ari, "Fast parallel algorithms for QR and triangular factorization", *SIAM J. Sci. Statist. Comput.*, 1987, pp. 899–913.

[7] J. Cioffi, "The fast adaptive ROTOR's RLS algorithm", *IEEE Trans. Acoust. Speech Signal Process.*, April 1990, pp. 631–653.

[8] G. Cybenko, "A general orthogonalization technique with applications to time series analysis and signal processing", *Math. Comput.* 1983, pp. 323–336.

[9] G. Cybenko, "The numerical stability of the lattice algorithm for least squares linear prediction systems", *BIT*, Vol. 24, 1984, pp. 441–455.
[10] G. Cybenko, "Fast Toeplitz orthogonalization using inner products", *SIAM J. Statist. Comput.*, 1987, pp. 734–739.
[11] G. Cybenko and M. Berry, "Hyperbolic householder algorithms for factorization of structured matrices", *SIAM J. Matrix Anal. Appl.*, Vol. 11, No. 4, October 1990, pp. 499–520.
[12] C. Demeure and L.L. Scharf, "Sliding windows and lattice algorithms for computing QR factors in the least squares theory of linear prediction", *IEEE Trans. Acoust. Speech Signal Process.*, April 1990, pp. 721–725.
[13] B. Friedlander, "Lattice filters for adaptive processing", *Proc. IEEE*, Vol. 70, August 1982, pp. 829–867.
[14] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1989.
[15] F. Itakura and S. Saito, "Digital filtering techniques for speech analysis and synthesis", in *Proc. 7th Internat. Congr. Acoust.*, Budapest, 1971, pp. 261–264.
[16] N. Kalouptsidis, G. Carayannis, D. Manolakis and E. Koukoutsis, "Efficient. recursive in order LS FIR filtering and prediction", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-33, October 1985, pp. 1175–1187.
[17] D. Lee, M. Morf and B. Friedlander, "Recursive least squares ladder estimation algorithm", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-29, June 1981, pp. 627–641.
[18] N. Levinson, "The Wiener RMS error criterion in filtering design and prediction", *J. Math. Phys.*, Vol. 25, 1947, pp. 261–278.
[19] F. Ling, "Givens rotation based least squares lattice and related algorithms", *IEEE Trans. Signal Process.*, Vol. SP-39, 1991, pp. 1541–1551.
[20] F. Ling, D. Manolakis and J. Proakis, "A recursive modified Gram–Schmidt algorithm for least-squares estimation", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-34, August 1986, pp. 829–836.
[21] J. Makhoul, "Linear prediction: A tutorial review", *Proc. IEEE*, Vol. 63, April 1975, pp. 561–580.
[22] J.D. Markel and A.H. Gray, *Linear Prediction of Speech*, Springer, New York, 1976.
[23] S.L. Marple, "Efficient least squares FIR system identification", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-29, 1984, pp. 62–73.
[24] S.L. Marple, *Digital Spectral Analysis and Applications*, Prentice-Hall, Englewood Cliffs, NJ, 1987.
[25] J.H. McClellan, "Parametric signal modeling", in: J. Lim and A. Oppenheim, eds., *Advanced Topics in Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1988.
[26] M. Morf, B. Dickinson, T. Kailath and A. Viera, "Efficient solution of covariance equations for linear prediction", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-25, 1977, pp. 429–433.
[27] I.K. Proudler, J.G. McWhirter and T.J. Shepherd, "Computationally efficient QR decomposition approach to least squares adaptive filtering", *IEE Proc. F*, Vol. 138, No. 4, August 1991, pp. 341–353.
[28] S. Qiao, "Hybrid algorithm for fast Toeplitz orthogonalization", *Numer. Math.*, Vol. 53, 1988, pp. 351–365.
[29] C.P. Rialan and L.L. Scharf, "Fast algorithms for computing the QR and Cholesky factors of Toeplitz operators", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-36, November 1988, pp. 1740–1748.
[30] C.P. Rialan and L.L. Scharf, "Fixed-point error analysis of the lattice and the Schur algorithms for the autocorrelation method of linear prediction", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-37, December 1989, pp. 1950–1987.
[31] D.R. Sweet, "Fast Toeplitz orthogonalization", *Numer. Math.*, Vol. 43, 1984, pp. 1–21.
[32] S. Theodoridis, "Pipeline architecture for block adaptive LS FIR filtering and prediction", *IEEE Trans. Acoust. Speech Signal Process.*, January 1990, pp. 81–90.
[33] B. Yang and J.F. Bohme, "Rotation-based RLS algorithms: Unified derivations, numerical properties and parallel implementations", *IEEE Trans. Signal Process.*, May 1992, pp. 1151–1167.